

LOCAL MAC ADDRESS LEARNING IN LAYER 2 FRAME FORWARDING

Brian M. Bass, Terri L. Davis, Anthony M. Gallo,
Michael S. Siegel, Fabrice J. Verplanken, Gail I. Woodland

CROSS REFERENCE TO RELATED APPLICATIONS

This application is related by common inventorship and subject matter to co-pending application titled "Method for Bridging Control Point Data Frames Using a Network Processor". Until such time as the foregoing application is assigned an application number by the U.S. Patent and Trademark Office, it may be referenced by the following applicant:attorney docket numbers: RAL9-00-0035, 421/061. The listed application is assigned to International Business Machines Corporation and is entirely incorporated herein by this reference.

BACKGROUND OF THE INVENTION

The present invention relates to computer networks such as LANs (local area networks), and more particularly to a software-based method for reducing network bandwidth consumption and economizing on database space in MAC (media access control) address learning.

In networks, data is typically exchanged between communicating devices in the form of "frames." Frames include a

source MAC address and a destination MAC address; a MAC address uniquely identifies a network device in a "Layer 2" communication protocol used in Ethernet and Token Ring LANs.

A network switch connected between communicating devices
5 uses the destination MAC address in an incoming frame generated by a source device to forward the frame to the appropriate target device. The switch typically includes a "control point" (CP) comprising a central processor and control software, and a plurality of forwarding processors. To facilitate forwarding, the CP and forwarding processors perform MAC address "learning."
10

MAC address learning refers to a process for reducing network traffic, wherein the source MAC address in a received frame is recorded in an address database along with its port of origin. This allows future frames destined for the source address to be forwarded
15 only to the port on which that address is located. Otherwise, an unrecognized address must be "flooded out", i.e., forwarded to every port where the address may reside.

In existing systems, the CP maintains a central database of learned MAC addresses. A number of distributed MAC address
20 databases which are duplicates of the central database are also maintained, locally to groups of ports and forwarding processors. As frames are received, a receiving forwarding processor performs look-ups of the source MAC addresses in the local databases, to determine whether they have been already learned. If not, the

forwarding processor notifies the CP, and in response, the CP learns the source address (i.e., records it in the central database along with its port of origin), and adds the address to every local database.

5 In a variation of the above, the local databases are not duplicated, but the CP is still used to learn source MAC addresses whenever a forwarding processor cannot find the address in a local database.

10 In another variation, the CP is not used in learning; instead, if a forwarding processor cannot find a source address in a local database, it sends control messages to all other forwarding processors, directing them to add the source address to their respective local databases.

15 It may be appreciated from the foregoing description that existing methods of MAC address learning entail considerable bandwidth overhead, due to the need for notifying the CP each time an unknown source address is received and for subsequently downloading the address to every local database, or for sending control messages. Also, there is unnecessary duplication of MAC address databases at distributed local sites; most local sites (i.e.,
20 physically separate groups of ports and forwarding processors) only use a small proportion of the overall MAC addresses.

In view of the foregoing, a more efficient method of MAC address learning is needed.

SUMMARY OF THE INVENTION

According to the present invention, MAC address learning is performed locally rather than by a central processor. In a preferred embodiment, a MAC address database which is local to a destination or target port for a data frame is updated by a local forwarding processor with the source address and port of origin of the data frame. Similarly, the responding data frame is learned locally, in the MAC address database corresponding to the ingress port for the initial data frame. Once the addresses are learned locally, frame forwarding between the corresponding devices can be performed more efficiently.

Local as opposed to centralized learning includes the advantages of eliminating the unnecessary duplication of a central database at local sites. Space is also conserved because each local database contains only those addresses needed and used. Processing cycles are offloaded from the CP, which no longer needs to maintain a central database or download addresses to local databases. Bandwidth that would otherwise have been needed for centralized MAC address learning is available for frame forwarding.

Additionally, MAC address "aging" may be performed locally. Aging refers to a process of deleting infrequently-used addresses from a MAC database to save space. Local aging also offloads the CP.

A method of reporting the locally-learned, distributed MAC addresses databases in an aggregate form to network users or a network management application is also disclosed.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows an example of a communication network;

Figure 2 shows components in a network switch;

Figure 3 shows a process flow in a method for local MAC address learning according to the present invention;

Figures 4A-4E illustrate the application of the method shown in Figure 3 in a specific example;

Figure 5 shows entries in a MAC database being traversed for database aging;

Figure 6 shows a process flow in aging;

Figures 7A and 7B show a process flow in a method for user-controlled MAC address reporting;

Figures 8A and 8B further illustrate MAC address reporting; and

Figure 9 shows examples of computer-usable media which may be used to store computer-executable instructions according to the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Figure 1 shows an example of a typical network configuration in

which the method of the present invention would find useful application. Network devices 101 such as personal computers, servers, terminals for data entry and display, printers and the like, are connected by a physical medium 102 such as twisted pair or coax cable to nodes such as bridges 103, switches 100, routers 104, and hubs/repeaters (not shown). Data generated by end users of the network devices travels across the network to other end users via the physical medium and the nodes, which perform the processing necessary to ensure the data arrives in a usable condition with the proper recipient.

For some applications, such as LANs, networks may be confined to a relatively small geographic area, such as a building or a campus. However, large numbers of smaller networks can be linked to form extensive composite networks including thousands of devices and nodes. As devices contend for a share of the network bandwidth, data collisions can result which degrade network performance. To reduce collisions, a network may be logically partitioned into "segments", for example segments 105 and 106, connected by nodes such as bridges, switches and routers.

In general, bridges and switches transfer data frames between segments, by filtering, flooding, or forwarding the data frames. Filtering refers to dropping or discarding a received data frame when processing of the MAC addresses according to a set of rules. Flooding, as described above, refers to forwarding a data frame to all

ports where a destination address may reside, when the address has not been learned and is thus unrecognized. Forwarding refers to sending an incoming data frame having a learned MAC address to the corresponding known port or ports.

5 As noted above, MAC addresses uniquely identify a network device in a Layer 2 communication protocol. The Layer 2 protocol may be viewed as occupying a level in a "protocol stack" of established, commonly-used communication standards, beginning with a Layer 1, representing the physical medium used to transfer data, followed by Layer 2, then by a Layer 3 and 4 and so on. Generally speaking, the layers above Layer 2 handle exchanges between network entities where the exchanges require more processing than at the lower levels.

10 Figure 2 shows a switch 100 in greater detail. The switch includes a CP 200 and a plurality (not limited to three) of cards 202-204, connected to the CP 200 and to each other via a pipe 201 over which frames and control messages flow and which may be implemented as a DASL crossbar switch or a bus. Each card includes a local MAC database 208, a plurality (not limited to four) of forwarding processors 207, a specialized processor 209 known as a Guided Tree Handler (GTH), and a plurality of ports 210. Each card further includes a storage medium such as a ROM or RAM 205 for storing computer-executable instructions, referred to herein as "picocode" 206, which when executed by a forwarding processor and

007740-692450

When a card containing a target port receives a frame forwarded to it from another card on the switch (block 305), local MAC address learning at the destination (i.e., learning on the target card) according to the present invention is performed as shown in
5 blocks 305-312. As shown in blocks 306 and 307, a local processor 207 performs a look-up in the local MAC address database 208 for the source address. If the source address is not found, the processor notifies the GTH 209 that there is a new address to be learned, or, if the port of origin of the source address has changed,
10 the processor notifies the GTH 209 that the address information must be updated.

As noted above, the GTH is a specialized processor for accessing and maintaining the local MAC address database. More specifically, MAC addresses may be stored in a database structure
15 known as a "Patricia tree" which enables fast searching and access. The tree entries are also chained together in a singly-linked circular list. While any processor may perform a search of the MAC address database, the GTH is used to add, delete or update database entries, or perform any actions which require following the entry chain. (The
20 GTH and the local processor can communicate without using any of the bandwidth used for forwarding frames.) Upon being notified by a local processor as described above, the GTH records the source MAC address of the forwarded frame in the local database, along with its port of origin, as shown in block 308.

Decision block 309 may be executed in order to accommodate the widely-used "Spanning Tree" function which is defined in the 802.1D communication standard. The Spanning Tree standard was developed to prevent network loops and is well-known in the art. In Spanning Tree, there are five basic port states: disabled, blocking, listening, learning and forwarding. Frames may be forwarded only when both the source and target ports are in the forwarding state. Accordingly, depending on whether the source port is in the forwarding state or not, the frame is either dropped (block 311), or forwarded out the target port and sent via the physical medium to the target network device (block 310).

Figures 4A-4E illustrate an example of the application of the method shown in Figure 3. In Figure 4A, a network device 101 having MAC address "addr1" generates a frame 400 having a Layer 2 field including a destination address (DA) of "addr2"; the source address (SA) is addr1. The frame is received by a port 210 (specifically, port x) on a card 202 including a local MAC database 208.

Since the local MAC database does not contain the destination address, the frame is flooded, and a copy is sent to the card 204 having the port (specifically, port y) connected to the target device, addr2, as shown in Figure 4B.

As shown in Figure 4C, the source address is not present in the local MAC database on card 204, so the source address and port of

addresses locally at a target port, as opposed to using a CP, are evident. Unnecessary duplication of databases is avoided, since each local database contains only those addresses needed and used. Further, bandwidth that would otherwise have been needed for MAC address learning using a CP is available for frame forwarding.

An additional benefit of local MAC address learning is realized in an "aging" process which occurs for MAC addresses. Because, as noted above, networks can be very extensive, MAC address databases can become very large. Accordingly, in the aging process, database entries corresponding to addresses which have not been accessed for a pre-determined period of time are purged in order to free up space.

According to the present invention, aging is performed locally. As shown in Figure 5, the switch hardware records an access to an address when a look-up of a source MAC address is performed on a MAC database 208 and an entry is found, by setting an "entry seen" bit in the entry. An aging task resets the seen bit periodically. An entry is considered to be expired if the entry seen bit has not been set since the last time the entry was visited by the aging task. Each GTH traverses the local database to delete expired entries.

A process flow for aging is illustrated in Figure 6. The MAC database entries are circularly linked in the order they are learned. Because MAC address databases can be extremely large, the GTH

traverses only a portion of the database in a single time interval, beginning with the entry following the last one it examined during the previous time interval. Each GTH has a timer which "pops" (generates a signal) when a defined timer value expires. Because
5 the number of entries fluctuates, the aging task computes a number of entries which it must examine between timer pops in order completely traverse the database within a user-defined period.

As shown in blocks 600-603, if an access has not been recorded for the entry since the last time the entry was examined, the
10 GTH deletes the entry. If an access has been recorded, the GTH clears the record of the access (resets the entry seen bit).

It can be seen that the foregoing process of local aging saves the bandwidth of CP communication that would be utilized in aging performed by the CP, and offloads considerable processing from the
15 CP.

The advantage of local aging is even more evident in "accelerated" aging. Accelerated aging is performed in response to a "Topology Change Notification", indicating a re-configuration of the network, received by the Spanning Tree protocol running on the CP.
20 Accelerated aging requires that at the exit of the accelerated aging period, the only addresses left in the MAC address databases are those that were accessed during the accelerated aging period.

According to the present invention, at the beginning of accelerated aging period, the GTH traverses the entire local

database and eliminates the records of any accesses. At the end of the accelerated aging period, the GTH deletes any entry which does not have a record of an access during the accelerated aging period.

5 Topology change notifications are common events which can result in purging of entire databases, causing extreme stress to switches that perform aging from the CP. Thus, it can be seen that performing aging locally significantly relieves the CP.

10 The present invention further provides a method of controllably reporting on the status of the local MAC address databases to users, if desired. The method compiles the distributed MAC databases into an aggregate database, which may be the required form for reporting to network management applications on the switch. It may also be used for console display for viewing by humans.

15 To provide for address reporting as needed, computer-executable instructions according to the present invention perform a "mirroring" method. Figures 7A and 7B show a process flow for the method, with Figure 7A showing steps executed by the CP, and Figure 7B showing steps executed by a GTH.

20 If an optional mirroring function is enabled, the CP will start a report timer having a timer value configurable by a user. After a period of time defined by the timer value elapses, the CP will send a Begin Report Request to each GTH, as shown in block 700. As shown in blocks 710 and 711 of Figure 7B, upon receipt of the request, each GTH bundles a pre-defined number of MAC addresses

into a Begin Report Reply and sends the bundle to the CP, recording the last address sent (this address cannot be deleted while it is recorded as the last address). To locate a starting address for the Begin Report Reply, a GTH will perform a search on a default MAC address which is always present in the database, for instance, the BPDU address (the BPDU address is defined by the 802.1D standard and is used by the Spanning Tree algorithm). This address acts as a logical beginning of a chain of MAC addresses stored in the database.

As shown in blocks 701-704 in Figure 7A, when the CP receives the Begin Report Reply, it will add each MAC address into a mirror database if the address is not already in the database. Each MAC address in the mirror database has a timer value associated with it. When an address is first entered, the value is positive. Every time the MAC address is reported again the timer value will be reset to the positive value.

After a subsequent period determined by the report timer, the CP will send a Continue Report Request to each GTH (block 705). As shown in blocks 712 and 713, when a GTH receives this request, it obtains the MAC address chained after the last MAC address it reported, bundles that address and a pre-determined number of the addresses following it into a Continue Report Reply which it sends to the CP. The GTH records the last address sent.

As shown in blocks 706 and 707, as the CP receives the

the GTHs form bundles 803 of a pre-determined number of MAC addresses and send them to the CP, recording a last address sent 804. The CP compiles an aggregate mirror database 802 from the bundles, which may be reported out to a user on a display screen, for example, or reported to a network manager.

As noted above, the mirroring function is optional and configurable by a user. A user can set the reporting time interval in accordance with the level of traffic on the switch so as to achieve a tolerable level of reporting interference.

Figure 9 illustrates examples of computer-executable media such as diskette 900, CD-ROM 901, magnetic tape 902 or hard disk 903 which may be used to store and transport computer-executable instructions according to the present invention. As is well understood, the instructions may be retrieved from the storage media and executed by a processor to effect the method of the invention. Typically, a forwarding processor 207 or GTH 209 executes the picocode 206. The CP 200 typically executes a high-level language such as "C" code 904.

The foregoing description of the invention illustrates and describes the present invention. Additionally, the disclosure shows and describes only the preferred embodiments of the invention, but it is to be understood that the invention is capable of use in various other combinations, modifications, and environments and is capable of changes or modifications within the scope of the inventive concept as

expressed herein, commensurate with the above teachings, and/or the skill or knowledge of the relevant art. The embodiments described hereinabove are further intended to explain best modes known of practicing the invention and to enable others skilled in the art to utilize
5 the invention in such, or other, embodiments and with the various modifications required by the particular applications or uses of the invention. Accordingly, the description is not intended to limit the invention to the form disclosed herein. Also, it is intended that the appended claims be construed to include alternative embodiments.

001170" 63E24560